Data Structures

Practical 6

Name: Utkarsh Pathak

EN20CS301480

Objective:

WAP to implement Stack data structure using Array.
Instruction: Submit in PDF format.

Code:

```
#include <iostream>

using namespace std;

int stack[10], n=10, top;

void initialize(){

top=-1;

cout<<"Stack is initialized"<<endl;

}

void overflow(){

if(top>n-1){

cout<<"Stack Overflow"<<endl;

}

else{

cout<<"Stack is not full"<<endl;

}

}
```

```
void underflow(){
if(top<=-1){
cout<<"Stack Underflow"<<endl;
}
else{
cout<<"Stack is not empty"<<endl;
}
}
void push(int val) {

if(top>=n-1)
cout<<"Stack Overflow"<<endl;
else {
top++;
stack[top]=val;
}
}
void pop() {
if(top<=-1)
cout<<"Stack Underflow"<<endl;
else {
cout<<"The popped element is "<< stack[top] <<endl;
top--;
}
}
void display() {
if(top>=0) {
cout<<"Stack elements are:"<<endl;
for(int i=top; i>=0; i--)
```

```
cout<<stack[i]<<endl;
cout<<endl;
} else
cout<<"Stack is empty"<<endl;
}
int main() {
int ch, val;
cout<<"1) Initialize Stack"<<endl;
cout<<"2) Underflow check"<<endl;
cout<<"3) Overflow check"<<endl;

cout<<"4) Push in stack"<<endl;
cout<<"5) Pop from stack"<<endl;
cout<<"6) Display stack"<<endl;
cout<<"7) Exit"<<endl;
do {
cout<<"Enter choice: ";
cin>>ch;
switch(ch) {
case 1:{
initialize();
cout<<endl;
break;
}
case 2:{
underflow();
cout<<endl;
break;
}
```

```
case 3:{
overflow();
cout<<endl;
break;
}
case 4: {
cout<<"Enter value to be pushed: ";
cin>>val;
push(val);
cout<<endl;
break;

}
case 5: {
pop();
break;
cout<<endl;
}
case 6: {
display();
cout<<endl;
break;
}
case 7: {
cout<<"Exit"<<endl;
break;
}
default: {
cout<<"Invalid Choice"<<endl;
```

```
}

}

}while(ch!=7);

return 0;

}
```

Output: